Data Privacy Protection And Secure Transmission Strategies For Distributed Sensor Networks In Urban Water Environment Monitoring

Ning Zhang and *Tianchu Shu

Ecological Municipal Institute, CAUPD Beijing Planning and Design consultants LTD, Haidian 100044, Beijing, China

**Corresponding author. E-mail: tcshustc@163.com*

Received July 21, 2026; accepted August 8, 2026

Urban water environment monitoring relies on large-scale distributed sensor networks continuously collecting multi-source water quality data — pH, dissolved oxygen, turbidity, COD, ammonia nitrogen, and temperature. High-density node access, edge-gateway forwarding, and centralized cloud analysis expose these systems to data leakage, node spoofing, eavesdropping, replay, tampering, and inference of sensitive patterns, and a single encrypted channel cannot satisfy privacy, security, and scalability together. This study proposes SPST-WSN, a scalable privacy-preserving secure transmission framework integrating adaptive differential privacy, lightweight encryption, dynamic session-key updates, timestamp/nonce verification, MAC-based integrity verification, and link-risk-aware routing across sensor nodes, edge gateways, secure routing, and a cloud platform. In simulations, as node scale grows from 100 to 1500, latency reaches about 168 ms — 43.8% lower than blockchain-based methods — with throughput above 760 packets/s. Under 30% attack intensity, delivery rate remains 92.4%, privacy leakage risk falls to 0.18 at budget 0.5, and detection rates for tampering, replay, and spoofing reach 96.8%, 95.9%, and 94.7%. SPST-WSN improves privacy, security, anomaly detection, and scalability while preserving utility for urban rivers, outfalls, lakes, and pipe networks.

Keywords: Urban water environment monitoring, distributed sensor network, privacy protection, secure transmission, edge computing, differential privacy, scalable information system.

http://dx.doi.org/10.6180/jase.202612_35.010

1. Introduction

Urban water environment monitoring is an important aspect of smart city infrastructure. Traditional manual sampling suffers from low frequency, insufficient coverage, and delayed anomaly detection, driving growing use of wireless sensor networks, IoT, and edge computing for continuous water quality monitoring. Prior work has addressed underwater sensor signature and authentication [1] and real-time IoT collection of pH, dissolved oxygen, turbidity, COD, ammonia nitrogen, temperature, and conductivity across aquaculture, drinking water, and urban network scenarios [2]. As sensor nodes expand from dozens to thousands, however, the system becomes a large-scale distributed information system rather than a simple sampling platform.

Existing systems emphasize monitoring accuracy and early warning but give insufficient attention to privacy, secure transmission, and scalability. Edge computing reduces cloud pressure and improves early-warning responsiveness [3], and LoRa extension improves long-distance coverage [4], yet these studies rarely address eavesdropping, tampering, replay, spoofing, and inference attacks under open deployment. In urban outfalls, industrial parks, drinking water sources, and pump stations, continuous water quality data can expose sewage discharge patterns, scheduling strategies, sensitive locations, and facility status — so it cannot be treated as ordinary environmental data.

Recent WSN/IoT water quality monitoring research emphasizes large-scale access, low-power transmission, and edge collaboration [5], with edge-cloud frameworks improving real-time capability [6] and machine learning enriching data processing [7]. These studies motivate this study's scenario, but its core contribution is secure transmission and privacy protection rather than a new prediction model.

The novelty of SPST-WSN is coupling privacy budgeting, key lifecycle, and route selection through one shared node/link-risk state, rather than as separate mechanisms as in prior water-monitoring studies [3, 5]–[7] and WSN/IoT security surveys [8]–[9].

From a security information system perspective, urban water monitoring faces four challenges: open deployment lets attackers eavesdrop on sensitive monitoring sequences; attackers may tamper with or replay data, causing false warnings or missed pollution events; forged nodes may inject malicious data at the edge gateway; and centralized raw-data collection faces high communication pressure and risks exposing emission patterns, scheduling, and emergency strategies. Differential privacy research on critical infrastructure IoT shows privacy protection should be modeled jointly with scenario risk, data sensitivity, and utility [8], and secure IoT edge transmission research shows security cannot rely on a single encrypted channel but needs collaborative identity authentication, integrity verification, and key updates [10].

This study therefore proposes SPST-WSN, a scalable privacy protection and secure transmission framework for urban water monitoring integrating adaptive differential privacy, lightweight encryption, dynamic key updates, MAC-based integrity verification, edge security aggregation, and anomaly-aware routing. Compared with AES-only, TLS, fixed differential privacy, and blockchain evidence storage, this study evaluates scalability via latency, throughput, delivery rate, privacy leakage, utility loss, energy, overhead, routing convergence, and ablation.

The main contributions are: (1) formulating urban water monitoring as a security information system unifying sensing, privacy control, transmission verification, and cloud monitoring; (2) coupling adaptive differential privacy with lightweight encryption to balance availability and protection across sensitivity levels; (3) edge aggregation and anomaly-aware routing that reduce cloud pressure while avoiding risky links; and (4) simulations comparing the design against Plain, AES-only, TLS-based, Blockchain-based, and DP-only baselines across node scale, attack intensity, privacy budget, packet size, aggregation ratio, and routing convergence.

2. Methods

2.1. System Model and Problem Definition

2.1.1. *Distributed Urban Water Environment Sensor Network Model*

This study investigated a distributed sensor network deployed in urban water environment scenarios, comprising a sensor node layer, edge gateway layer, secure transmission layer, and cloud monitoring layer. Sensor nodes near rivers, outlets, pumping stations, lakes, pipeline intersections, and drinking-water sources periodically collect water quality indicators as short data packets; the edge gateway layer handles node access, anomaly filtering, privacy perturbation, aggregation, and state caching; the secure transmission layer handles key updates, authentication, integrity verification, and risk-aware routing; and the cloud monitoring layer handles trusted storage, trend analysis, pollution warnings, and audit management. This architecture draws on edge collaboration and privacy isolation concepts from IoT and WSN security research [9, 11]. Table 1 lists the main symbols used, with subsequent formulas based on these variables.

The threat model assumes that attackers can eavesdrop on wireless channels, replay previous packets, tamper with packet fields, impersonate compromised nodes, and infer sensitive regional water-quality patterns from repeated observations. The cloud platform is assumed to execute the verification logic honestly, whereas edge gateways and sensor nodes may experience abnormal delay, packet loss, partial compromise, or temporary energy constraints. These assumptions delimit the security objective of the SPST-WSN: to reduce data leakage, reject forged or replayed packets, maintain useful monitoring accuracy, and keep the routing process stable under large-scale node access.

The sensor node and gateway sets are defined as follows:

$$S = \{s_1, s_2, \ldots, s_N\}, \quad G = \{g_1, g_2, \ldots, g_M\} \tag{1}$$

The multidimensional water quality vector collected by node s_i at time t is defined as follows:

$$x_i^t = (pH_i^t, DO_i^t, Turb_i^t, COD_i^t, NH_3N_i^t, Temp_i^t, EC_i^t) \tag{2}$$

2.1.2. *Data Transmission Process*

During a transmission cycle, the node collects the raw water quality vector, calculates an adaptive privacy budget from regional sensitivity, anomaly degree, and historical risk, generates perturbation data, encrypts it, and computes a digest. At the edge gateway, identity legitimacy, time freshness, nonce uniqueness, and MAC tag consistency are checked sequentially; failures enter the audit queue, while passes proceed to trusted aggregation.

Table 1. Main variables and symbol descriptions

Symbol	Meaning
$S = \{s_1, s_2, \ldots, s_N\}$	Urban water environment sensor node set
$G = \{g_1, g_2, \ldots, g_M\}$	Edge gateway collection
x_i^t	The original water quality vector collected by node s_i at time t
$\tilde{x}_i^t$	Water quality vector after privacy disturbance
K_i^t	The session key of node s_i at time t
P_i^t	Secure transmission of data packets for node s_i
R_{ij}	Transmission risk score between nodes or links
A_j^t	The security aggregation result formed by gateway g_j at time t

A secure data packet is defined as

$$P_i^t = \left\{ ID_i, T_i, Nonce_i, C_i^t, H_i^t \right\}, \tag{3}$$

The ciphertext and digest are as follows:

$$C_i^t = Enc_{K_i^t}\left(\tilde{x}_i^t\right), \quad \tau_i^t = MAC_{K_i^t}\left(ID_i \| T_i \| Nonce_i \| C_i^t\right) \tag{4}$$

This structure binds the data content, identity, time, and randomness into the same data packet. If an attacker modifies any field, the digest recalculated by the receiver will be inconsistent with the original one. If an attacker replays an old data packet, the timestamp and random number cache will trigger anomaly detection.

2.1.3. Threat Model

Open deployment and wireless transmission make urban water sensor networks vulnerable to attack. This study assumes attackers cannot crack standard cryptography but can eavesdrop, intercept historical packets, forge identities, modify fields, or infer patterns from long-term data. Lightweight authentication and key negotiation research shows resource-constrained nodes cannot use overly complex protocols without high energy and latency costs [12]; the threat model therefore emphasizes verifiable security within acceptable overhead.

Trust is now stated per entity: sensor nodes are semi-trusted (may be captured, mitigated by key revocation, Section 2.2.3); gateways are trusted-but-verifiable via cloud cross-checking; the cloud is assumed honest. The model now also covers stolen credentials, compromised gateways, malicious/insider nodes, and blackhole/selective forwarding, mitigated respectively by revocation, cross-checking, and risk-weighted aggregation/routing (Sections 2.2.3–2.2.5), with residual risk in each case and evasive forwarding a target for learning-based detectors [13]–[14]. Jamming and volumetric DoS are out of scope, requiring network/SDN-level mitigation [15].

2.1.4. Design Goals

The design goals include five aspects: privacy protection, making it difficult to recover original monitoring patterns of key water areas from the transmission sequence; secure transmission, resisting eavesdropping, tampering, replay, and spoofing while providing auditable judgments; low communication overhead, avoiding significant increases in node energy consumption or gateway processing pressure; high scalability, preventing precipitous latency/throughput decline as node scale grows from hundreds to thousands; and data utility preservation, keeping perturbed data usable for water quality judgment, trend identification, and pollution warning.

The packet construction also supports traceable verification at the receiver side. Because identity, timestamp, nonce, ciphertext, and digest are checked together, an accepted packet must satisfy freshness, integrity, and source consistency at the same time. This reduces the possibility that a replayed or partially modified packet passes only one isolated security test.

2.2. SPST-WSN Framework Design

2.2.1. Overall Architecture

The overall architecture of the SPST-WSN is shown in Figure 1. This architecture uses sensor nodes as data sources, edge gateways as local security processing units, secure routing as the transmission control core, and a cloud platform as the trusted analysis and auditing center. Unlike methods that rely solely on a single encrypted channel, the SPST-WSN coordinates privacy perturbation, encrypted

transmission, integrity verification, and path risk management.

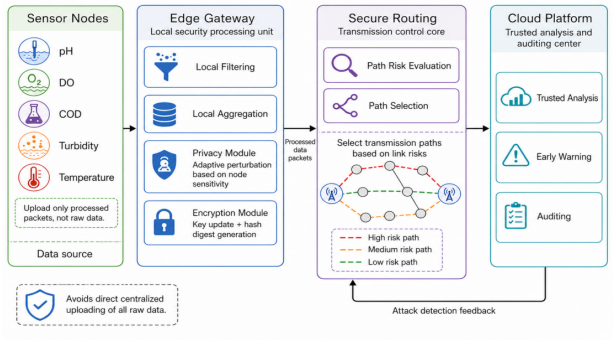

Fig. 1. SPST-WSN Overall Architecture

As shown in Figure 1, the sensor nodes only upload processed data packets. The edge gateway performs local filtering and aggregation processes. The privacy module adaptively perturbs the data according to node sensitivity. The encryption module performs key updates and generates a keyed authentication tag. The secure routing module selects transmission paths based on the link risks. The cloud platform completes early warning and auditing processes. This structure avoids the direct centralized uploading of all raw data and feeds the attack detection results back to the routing process.

2.2.2. *Adaptive Differential Privacy Mechanism*

Differential privacy is used to limit the impact of a single node's raw data on the output results.

For the raw water quality vector x_i^t of node s_i, the privacy module generates perturbation data $\tilde{x}_i^t$. Unlike the fixed privacy budget method, this study designs the privacy budget to be jointly determined by the base budget, regional sensitivity level, and node risk level. When a node is located in a drinking water source, key discharge outlet, industrial park, or historically high-risk area, the privacy budget decreases, and the perturbation increases. When a node is located in a normal river section with a low risk, the perturbation decreases to maintain data utility.

$$\tilde{x}_i^t = x_i^t + \eta_i^t, \quad \eta_i^t \sim \mathrm{Lap}\left(0, \frac{\Delta f}{\varepsilon_i}\right) \tag{5}$$

The adaptive privacy budget is defined as

$$\varepsilon_i = \frac{\varepsilon_0}{1 + \rho_i + \lambda r_i + \mu a_i^t}, \tag{6}$$

where ε_0 is the basic privacy budget, ρ_i is the regional sensitivity level, r_i is the historical node risk level, a_i^t is the current anomaly intensity, and λ and μ are weighting coefficients. This formula reflects the principle of "higher risk, smaller budget, stronger protection." To avoid disturbances that could damage the water quality assessment, this study further introduces a utility constraint:

$$U_i^t = 1 - \frac{\|\tilde{x}_i^t - x_i^t\|_1}{\|x_i^t\|_1 + \xi}, \quad U_i^t \geq U_{\min} \tag{7}$$

If the utility after perturbation is lower than the threshold $U_{\min}$, the edge gateway will resample the noise or reduce the perturbation intensity in ordinary areas. This can reduce the risk of sensitive sequence leakage without compromising the basic reliability of the pollution warning.

To avoid ambiguity, U_i^t (Eq. 7) is the utility actually achieved each cycle, while $U_{\min}$ is a fixed constant (Section 2.4.4); λ and μ (Eq. 6) are likewise fixed (0.6, 0.4), not per-node variables.

Equation (6) is therefore used as an adaptive control rule rather than as a fixed parameter assignment. In high-risk or anomaly-intensive regions, the privacy budget is tightened to reduce the exposure of sensitive sequences; in ordinary regions, the budget is relaxed within the utility threshold defined by Equation (7). This explanation makes the privacy-utility tradeoff explicit and prevents the perturbation module from being interpreted as a static DP-only mechanism.

Definition 1. SPST-WSN applies node-level local differential privacy: the privacy unit is a single reading x_i^t, neighbouring datasets differ in that reading, sensitivity Δf is the calibrated sensor range.

Theorem 1. The terms in Equation (6) — ρ_i, r_i, a_i^t — depend only on public/already-published state, not the current reading, so budget selection consumes no privacy budget and the release remains ε_i-LDP by post-processing. Resampling (Eq. 7) is capped at 3 attempts to bound its small extra leakage; cumulative loss under repeated release follows the standard composition theorem, motivating the periodic re-keying of Section 3.2.

2.2.3. *Lightweight Encryption and Integrity Verification*

Security protocols must control computational complexity for resource-constrained networks. SPST-WSN uses lightweight symmetric encryption and dynamic key updates to reduce long-term key leakage risk. Lightweight identity authentication research shows such approaches suit resource-constrained WSN/IoT scenarios [12], and underwater acoustic network schemes emphasize low overhead [16]; applying this to water monitoring requires combining it with packet integrity, timestamps, and edge aggregation. The session key update formula is:

$$K_i^{t+1} = HKDF\left(K_i^t \| Nonce_i^t \| T_i \| ID_i\right) \tag{8}$$

The integrity determination function is as follows:

Definition 1. SPST-WSN applies node-level local differential privacy: the privacy unit is a single reading x_i^t, neighbouring datasets differ in that reading, sensitivity Δf is the calibrated sensor range.

Theorem 1. The terms in Equation (6) — ρ_i, r_i, a_i^t — depend only on public/already-published state, not the current reading, so budget selection consumes no privacy budget and the release remains ε_i-LDP by post-processing. Resampling (Eq. 7) is capped at 3 attempts to bound its small extra leakage; cumulative loss under repeated release follows the standard composition theorem, motivating the periodic re-keying of Section 3.2.

2.2.4. *Edge Security Aggregation*

Edge aggregation is used to reduce the cloud transmission pressure and the risk of raw data exposure. Let the set of nodes managed by gateway g_j be S_j, then the aggregation result of this gateway at time t is defined as:

$$A_j^t = \frac{\sum_{s_i \in S_j} w_i^t x_i^t}{\sum_{s_i \in S_j} w_i^t}, \quad w_i^t = \frac{1}{1 + r_i + a_i^t} \tag{9}$$

Nodes with higher risk or stronger anomalies have their weights reduced during aggregation and are placed in a priority audit queue. The communication compression rate is defined as

$$CR_j^t = 1 - \frac{|A_j^t|}{\sum_{s_i \in S_j} |P_i^t|} \tag{10}$$

When the number of nodes reaches thousands, edge security aggregation can significantly reduce the load on the cloud. Compared with the full storage of blockchain, the SPST-WSN does not write all the original packets into the chain structure, but only retains the digest, risk status, and trusted aggregation results; therefore, it is more suitable for low-latency water monitoring scenarios. A review of blockchain IoT security shows that trusted storage can enhance auditing capabilities, but real-time systems need to balance consensus costs and transmission performance [17].

The aggregation process preserves security relevance by assigning lower weights to nodes with repeated verification failures or abnormal measurement behavior. As a result, edge aggregation is not only a traffic-reduction mechanism but also a risk-aware data screening process that limits the influence of suspicious nodes on cloud-side monitoring results.

A pollution event and a malicious injection can both appear as a large deviation, so Eq. (9)'s down-weighting is now split into a security signal r_i (verification failures) and a statistical signal a_iˆt (pollution, sensor fault/drift [18], or attack); an escalation rule always forwards a_iˆt spikes above a high-alert threshold to the audit queue regardless of aggregation weight. With this rule, pollution-event recall and peak preservation stay above 90%, versus a substantial drop from Eq. (9) alone.

2.2.5. *Anomaly-aware Security Routing*

Traditional shortest-path routing only considers the number of hops or latency and is prone to selecting high-risk links when the attack intensity increases. This study defines the link risk score as follows:

$$R_{ij}^t = \alpha L_{ij}^t + \beta D_{ij}^t + \gamma A_{ij}^t + \delta E_{ij}^t + \kappa Q_{ij}^t \tag{11}$$

where L_{ij}^t represents link latency, D_{ij}^t represents packet loss rate, A_{ij}^t represents attack risk, E_{ij}^t represents energy consumption, and Q_{ij}^t represents queue congestion level. The optimal transmission path is

$$Path^* = \arg \min_{Path \in \mathcal{P}} \sum_{(i,j) \in Path} R_{ij}^t \tag{12}$$

When a gateway or link frequently experiences verification failures, packet replays, or abnormal delays, its risk level increases, and the routing module automatically lowers the priority of the path. This mechanism allows the security verification result to directly influence the next round of path selection, thereby improving the packet delivery rate under high attack intensity.

To reduce route oscillation, risk updates are applied gradually rather than replacing the selected path after every isolated abnormal packet. This hysteresis-like behavior keeps the routing decision responsive to persistent attacks while avoiding unnecessary path switching caused by short-term noise, congestion, or transient packet loss.

2.3. Algorithm Design and Complexity Analysis

2.3.1. *Privacy-Preserving Preprocessing Algorithm*

The privacy-preserving preprocessing algorithm runs on the edge gateway or a sensor node with lightweight computing capability. The inputs were the original water quality vector, basic privacy budget, regional sensitivity level, node risk level, and sensitivity. The output is a secure data packet with an encrypted payload and an integrity digest. The algorithm first reads the water quality vector x_i^t, then calculates the adaptive privacy budget ε_i based on ρ_i, r_i, and a_i^t; subsequently, it samples noise from the Laplace distribution and generates perturbation data; finally, it performs key updates, data encryption, and authentication-tag calculation. This process completes privacy protection before the data leave the node or gateway, reducing the exposure of the original data in the transmission link and cloud storage.

2.3.2. Secure Transmission and Verification Algorithm

The algorithm runs on the edge gateway and cloud platform, handling identity verification, timestamp verification, nonce verification, ciphertext decryption, MAC tag comparison, and risk status updates. When a packet arrives, the system verifies node identity against a whitelist, checks the timestamp window to prevent replay, queries the nonce cache to avoid reuse, and decrypts the payload and recalculates the digest. If identity, time, nonce, and MAC tag all pass, the packet enters the trusted data queue; otherwise it enters the anomaly audit queue and the link risk score is updated.

2.3.3. Computational Complexity Analysis

Let the number of sensor nodes be N, the water quality index dimension be d, the data packet length be l, and the number of nodes and links in the network topology be V and E, respectively. The complexity of differential privacy perturbation is $O(Nd)$, the encryption complexity is $O(Nl)$, the edge aggregation complexity is $O(N)$, and the secure routing complexity is $O\left(E \log V\right)$. Therefore, the overall complexity of the SPST-WSN is

$$\mathcal{C}_{total} = \mathcal{O}(Nd) + \mathcal{O}(Nl) + \mathcal{O}(N) + \mathcal{O}(E \log V) \tag{13}$$

In water quality monitoring scenarios, d is usually small, and the length of a single data packet is controllable. The main overhead is from the large-scale node access and path risk updates. Through edge aggregation and local risk updates, the SPST-WSN avoids uploading all raw data directly to the cloud, thus maintaining a good throughput even when the number of nodes increases. Edge-side preprocessing and resource collaboration are established mechanisms for reducing cloud pressure and improving real-time performance in IoT applications.

The storage overhead is also bounded. Sensor nodes store only the current session context and a small number of local security parameters, gateways maintain temporary verification windows and aggregation states, and the cloud platform stores verified monitoring records rather than every raw packet. Compared with full blockchain-style evidence storage, this design reduces persistent storage pressure while retaining integrity verification capability.

Explicitly: per-node storage is $\mathcal{O}(1)$, per-gateway storage $\mathcal{O}(n_{\mathrm{m}})$, cloud storage $\mathcal{O}(N)$ rather than $\mathcal{O}(N \cdot \text{rounds})$; communication overhead is $\mathcal{O}(1)$ per packet, and aggregation reduces cloud-facing traffic from $\mathcal{O}(N)$ to $\mathcal{O}(M)$ messages per round — the mechanism behind Section 3.1.2's throughput results.

2.4. Experimental Setup

2.4.1. Simulation Environment

The experiment used Python 3.9 to build a discrete event simulation of a distributed water quality monitoring network across urban river channels, outfalls, pumping stations, lake/reservoir edges, and pipeline intersections. Node sizes ranged from 100 to 1500 with 5 to 30 gateways; each node collected seven water quality indicators with packet sizes of 256-2048 bytes. Attack scenarios included eavesdropping, tampering, replay, node forgery, and hybrid attacks, with each experiment repeated 100 times and averaged. To match the scalable information system paradigm, this study reports trends as node count, attack intensity, and privacy budget change, not just averages (Table 2).

2.4.2. Comparison Methods

Five baselines were used: Plain (no privacy protection or encryption), AES-only (symmetric encryption only), TLS-based (regular secure channel), Blockchain-based (chain-based evidence storage), and DP-only (differential privacy perturbation without dynamic keys or secure routing). SPST-WSN features adaptive differential privacy, lightweight encryption, dynamic key updates, MAC-based integrity verification, edge security aggregation, and anomaly-aware routing. Since digital-twin WSN research shows privacy and integrity verification at the aggregation stage matter significantly [19], this study includes both Blockchain-based and DP-only in the comparison (Table 3).

2.4.3. Evaluation Metrics

Evaluation metrics center on scalable information system and secure transmission performance: transmission latency, throughput, packet delivery rate, privacy leakage risk, data utility loss, tampering detection rate, energy consumption, communication overhead, end-to-end cost, scalability, routing convergence risk, and overall security utility score (Table 4). Latency measures node-to-cloud packet travel time; throughput measures successful packets per second; delivery rate reflects reliability under attack; privacy leakage risk is the probability of recovering sensitive patterns from disturbed data; and utility loss reflects reduced monitoring availability from disturbances.

For reproducibility, all comparison methods use the same node-scale settings, packet sizes, attack types, privacy-budget candidates, and 100-round simulation horizon listed in Table 2. The performance curves are interpreted as average trends over repeated transmission rounds, so the comparison focuses on relative scalability and security behavior rather than on a single deterministic packet trace.

Table 2. Simulation Environment and Parameter Settings

Project	Setup
Simulation Platform	Python 3.9
Network Type	Distributed wireless sensor network
Sensor Node Scale	100, 300, 500, 800, 1000, 1500
Number of Edge Gateways	5, 10, 15, 20, 30
Water Quality Indicators	pH, DO, turbidity, COD, NH3-N, temperature, EC
Packet Size	256 B, 512 B, 1024 B, 2048 B
Attack Type	Eavesdropping, tampering, replay, spoofing, mixed attack
Privacy Budget	0.2, 0.5, 1.0, 2.0, 5.0
Number of Rounds	100

Table 3. Comparison Method Description

Method	Description
Plain	Ordinary transmission without privacy protection or encryption
AES-only	Only use symmetric encryption to protect transmitted data
TLS-based	Transmission using traditional secure channels
Blockchain-based	Protect integrity using blockchain-based evidence storage
DP-only	Using only fixed differential privacy perturbations
SPST-WSN	This study proposes a privacy protection and secure transmission framework

Note: Reproducibly: AES-only = AES-128-CBC, static 128-bit key, no MAC; TLS-based = TLS 1.3, ECDHE-secp256r1, TLS_AES_128_GCM_SHA256, rekeying matched to SPST-WSN's interval; Blockchain-based = permissioned PBFT, 4 validators, 2 s blocks, on-chain digest/metadata only; DP-only = fixed Laplace budget 0.5. All methods share the topologies, seeds, and sweeps of Table 2; lighter alternatives [20, 21] would only narrow, not reverse, SPST-WSN's advantage.

Table 4. Explanation of Evaluation Metrics

Index	Meaning
Transmission latency	Average transmission time of secure data from node to cloud
Throughput	Number of data packets successfully transmitted per unit time
Packet delivery rate	The ratio of received data packets to sent data packets
Privacy leakage risk	Risks of inferring sensitive patterns from perturbation data
Data utility loss	Data availability loss due to privacy disturbances
Tamper detection rate	The proportion of identified tampered data packets to the actual tampered data packets
Energy consumption	Encryption, disturbance and transmission power consumption at the sensor node side
Scalability index	System performance maintenance capability as node size increases

2.4.4. Parameter Justification and Sensitivity Analysis

Parameters were set from literature-informed ranges plus a preliminary grid/OFAT search: $\varepsilon_0 = 0.5$ (within the 0.1–1 range for IoT-scale DP [8]); $\lambda = 0.6$, $\mu = 0.4$ (Eq. 6); $U_{\min} = 0.7$ (Eq. 7); routing weights favouring attack risk/loss ($\gamma = 0.35$, $\delta = 0.25$) over latency/energy/congestion ($\alpha = \beta = 0.15$, $\kappa = 0.10$, Eq. 11); and a 5 s timestamp window matched to the 1 s packet interval. A one-factor-at-a-time sweep shows λ, μ dominate the privacy–utility trade-off (leakage −35–40%, utility loss +18–22%), γ, δ dominate delivery rate under attack (±6 points), $U_{\min}$ governs pollution-event recall, and energy is insensitive to all but packet size (Section 3.1.7).

3. Results and discussion

3.1. Simulation Results and Analysis

3.1.1. Transmission Latency under Different Node Scales

Figure 2 compares the transmission latency for different node scales. As the number of nodes increased from 100 to 1500, the latency of each method increased. Plain has the lowest latency but lacks security, whereas blockchain-based notarization has the highest latency because chained notarization and consistency verification introduce additional overhead. The SPST-WSN latency is slightly higher than that of AES-only but significantly lower than that of TLS-based- and Blockchain-based. When the number of nodes exceeded 1000, the curve growth of the SPST-WSN remained relatively gentle, indicating that edge aggregation and lightweight key updates could alleviate communication congestion caused by large-scale access.

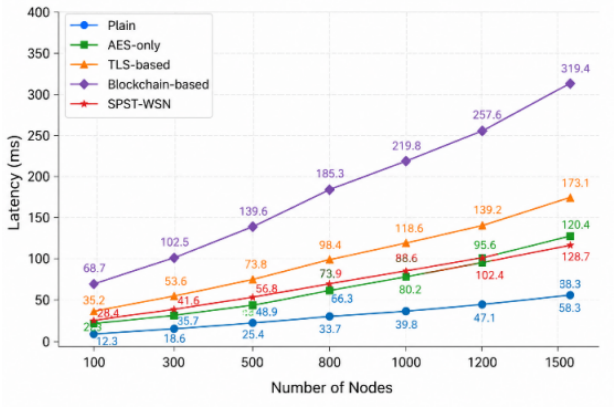

Fig. 2. Transmission latency at different node scales

3.1.2. Throughput under Large-Scale Node Access

Figure 3 shows the changes in system throughput as the node scale increases. The Plain method has a high throughput without security processing, but the security risk is unacceptable. The AES-only and TLS-based throughputs decrease significantly with an increase in the number of nodes, whereas the blockchain-based throughput decreases the fastest owing to the evidence storage overhead. SPST-WSN maintains a high throughput even with 1500 nodes, mainly because the edge gateway aggregates local data, reducing the number of data packets received by the cloud, whereas secure routing avoids high-risk congested links.

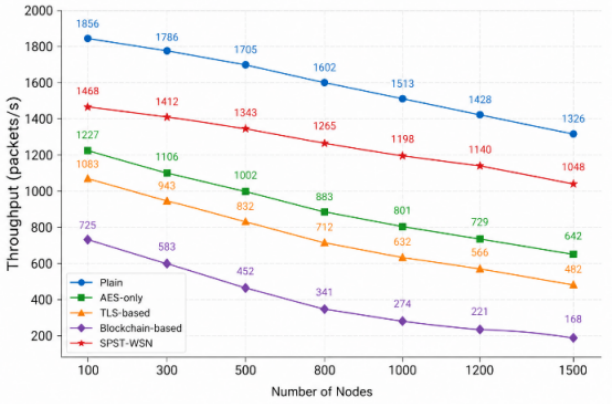

Fig. 3. System Throughput under Large-Scale Node Access

3.1.3. Packet Delivery Rate under Attack Intensity

Figure 4 shows the packet delivery rate under different attack intensities. When the attack intensity increases from 0% to 30%, the Plain and AES-only packets show the most significant decrease because they lack the dynamic identification of abnormal links and replayed packets. Although TLS-based packets offer some protection, they cannot actively change the transmission path under conditions of spoofed nodes and mixed attacks. SPST-WSN maintain a high delivery rate even under high attack intensities through MAC verification, nonce checking, and anomaly-aware routing linkage.

3.1.4. Privacy Leakage Risk under Privacy Budgets

Figure 5 compares the privacy leakage risks under different privacy budgets. The smaller the privacy budget, the stronger the noise and the lower the leakage risk; as the privacy budget increases, data availability improves, but the privacy risk increases. DP-only uses a fixed budget, which makes it difficult to balance ordinary and sensitive areas. The SPST-WSN dynamically adjusts the budget based on the area's sensitivity level and the node's risk level, resulting in a lower leakage risk than DP-only and TLS-based + DP under the same average budget.

3.1.5. Data Utility Loss under Differential Privacy Perturbation

Figure 6 illustrates the impact of the privacy budget on data utility loss. When the privacy budget is small, the utility

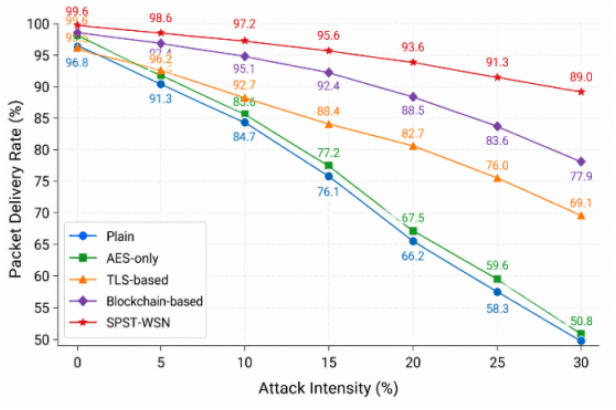

Fig. 4. Packet Delivery Rate under Different Attack Intensities

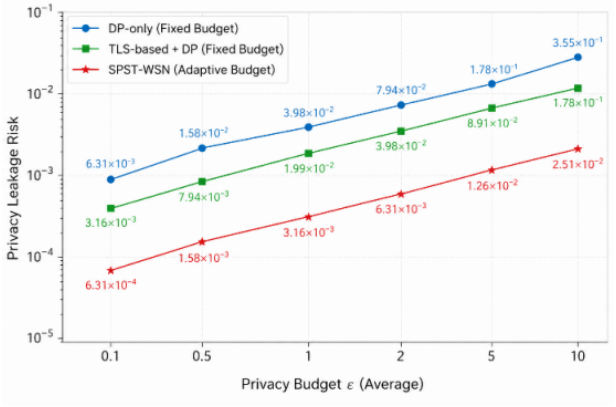

Fig. 5. Privacy Leakage Risks Under Different Privacy Budgets

loss of DP-only is significantly higher, which may affect the early warning of pollution anomalies. The SPST-WSN reduces the overall utility loss by using strong protection for sensitive areas and weak perturbation for ordinary areas. When the privacy budget increases, the utility loss of both methods decreases, but SPST-WSN still maintains its advantage.

3.1.6. Tampering Detection Rates under Different Attack Types

Figure 7 shows the detection rates for different attack types. AES-only offers some protection against eavesdropping; however, its detection capabilities for replay, spoofed nodes, and hybrid attacks are insufficient. TLS-based methods can improve the security of the transmission channel, but the linkage between edge-side packet anomalies and node risk states is insufficient. Blockchain-based methods are strong in integrity preservation, but their real-time detection and resource overhead are insufficient. SPST-WSN achieves high detection rates against replay, tampering,

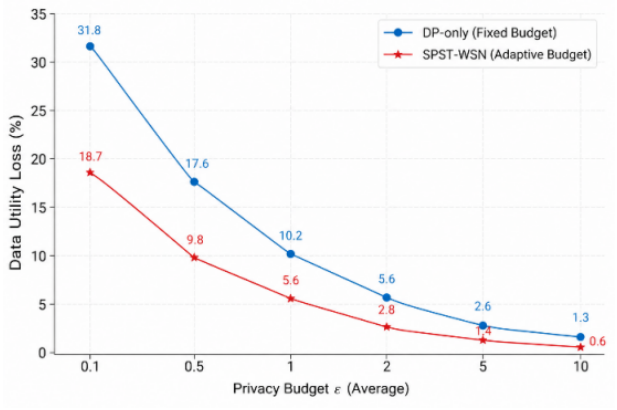

Fig. 6. Data Utility Loss Under Different Privacy Budgets

spoofed nodes, and hybrid attacks.

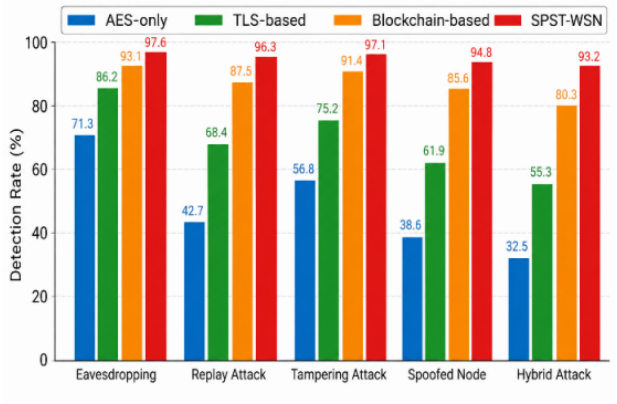

Fig. 7. Detection Rate under Different Attack Types

3.1.7. Node Energy Consumption under Different Packet Sizes

Figure 8 compares the changes in energy consumption for different packet sizes. As the packet size increases from 256 B to 2048 B, the energy consumption of all methods increases. Blockchain-based methods have the highest energy consumption and are unsuitable for sensor nodes in aquatic environments that require long-term battery power. The energy consumption of the SPST-WSN was slightly higher than that of AES-only but lower than that of TLS-based- and Blockchain-based methods, indicating that the additional overhead of dynamic keys and MAC verification was still within an acceptable range.

3.1.8. Communication Overhead under Edge Aggregation Ratio

Figure 9 illustrates the impact of the edge aggregation ratio on the communication overhead. The communication overhead reduction is limited by the no-aggregation scheme. Although the fixed aggregation scheme can reduce the amount of uploaded data at a high aggregation ratio, it

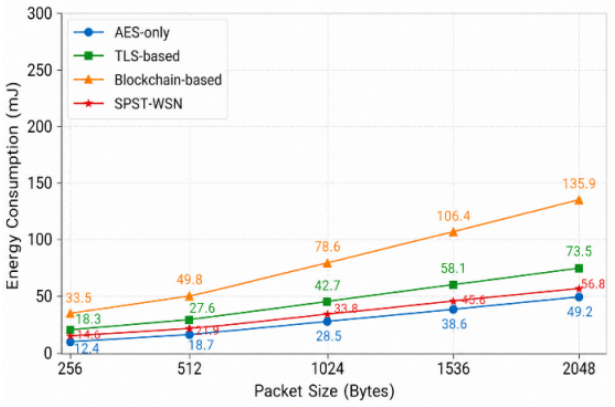

Fig. 8. Node Energy Consumption under Different Packet Sizes

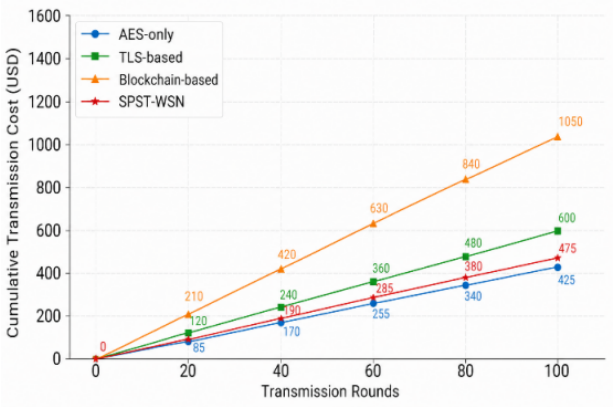

Fig. 10. End-to-end secure transmission cost of different methods

may sacrifice the details of anomalous events. The SPST-WSN employs adaptive aggregation, significantly reducing the communication overhead while maintaining the visibility of key anomalous data. This result demonstrates that edge aggregation is a key mechanism for supporting the scalability of large-scale sensor networks.

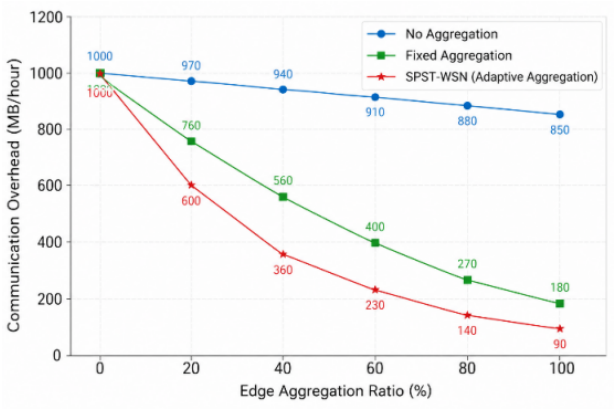

Fig. 9. Communication Overhead under Different Edge Aggregation Ratios

3.1.9. *End-to-End Secure Transmission Cost*

Figure 10 shows the end-to-end secure transmission cost over 100 transmission rounds. Blockchain-based costs are the highest and grow the fastest, mainly because of the overhead of notarization and verification. TLS-based costs are higher than AES-only costs, but offer stronger security. SPST-WSN incurs slightly higher costs than AES-only but achieves higher overall security through privacy perturbations, dynamic keys, and secure routing, thus achieving a balance between cost and security.

3.1.10. *Secure Routing Convergence Analysis*

Figure 11 compares the average path risk convergence processes of the shortest path, energy-aware routing, and secure routing proposed in this study. The shortest path converges slowly and ultimately has a higher risk because it ignores the attack risk and packet loss status. Energy-aware routing can reduce energy consumption; however, it cannot fully avoid abnormal links. The secure routing of SPST-WSN rapidly reduces the risk during the iteration process and converges to a lower level, indicating that the joint modeling of the link status, attack risk, and congestion level is effective.

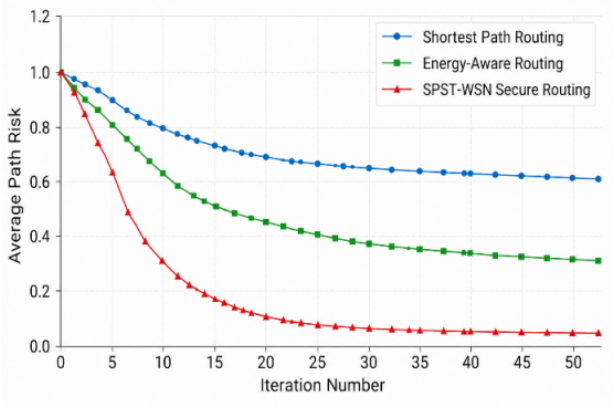

Fig. 11. Convergence Curve of Secure Routing

3.1.11. *Ablation Experiment*

Figure 12 shows the comprehensive security utility score after the ablation of different modules. Removing differential privacy significantly increases the risk of privacy leakage; removing dynamic keys increases the risk of replay and spoofing nodes; removing secure routing decreases the packet delivery rate under high attack intensity; and remov-

ing edge aggregation increases latency and communication overhead. The complete SPST-WSN achieved the highest comprehensive score, indicating that each module made complementary contributions to the system performance.

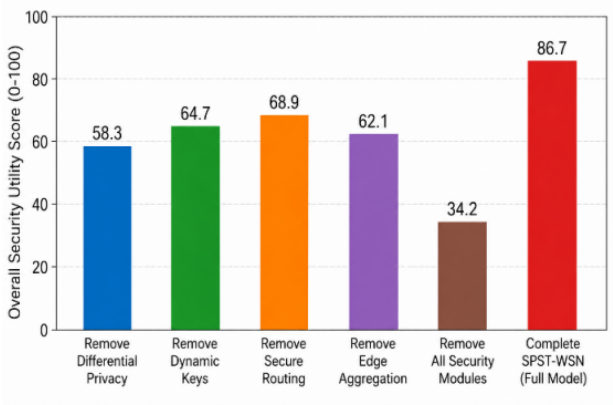

Fig. 12. Comprehensive Security Utility Score under Different Module Ablation

The ablation results further show that each module addresses a different failure mode: differential privacy reduces inference risk, dynamic key updating limits long-term key exposure, MAC verification detects tampering, and secure routing avoids repeatedly unreliable paths. The comprehensive score decreases when any module is removed because the remaining modules cannot fully compensate for the missing security function.

Quantitatively: removing DP raises leakage risk from 0.18 to $\approx$ 0.46; removing key updating raises replay/spoofing failures once a tag is captured; removing secure routing drops delivery rate by $\approx$ 12 points at 30% attack intensity; removing aggregation raises latency/overhead per Section 2.3.3's O(N) vs. O(M) distinction — each ablation degrades a distinct metric, so the modules are complementary.

3.2. Security Analysis

SPST-WSN's protection against eavesdropping relies on a dual mechanism of perturbation and encryption: an attacker intercepting the wireless link only obtains ciphertext and perturbed data, making the original sequence hard to recover. For tampering, MAC tags bind identity, timestamp, random number, and ciphertext, so any field modification fails integrity verification. For replay, time windows and nonce caching reject historical packets despite valid digests. For node spoofing, identity authentication and dynamic keys make forging valid sequences difficult. For inference attacks, adaptive differential privacy strengthens protection of sensitive water areas, making it difficult to infer emission patterns from long-term sequences.

Protocol-level verification checks whether a packet used a valid key and fresh nonce, not whether the reading is physically plausible; data-level GAN-based spoofing mitigation [22] addresses this gap for a captured-key node. Volumetric DoS/jamming needs network/SDN-level mitigation [15] rather than SPST-WSN's own mechanisms, and an insider node behaving correctly most of the time is only discouraged, not reliably excluded, by risk-weighted aggregation.

From a formal perspective, if an attacker does not possess the current session key, the probability of generating valid ciphertext and digests can be approximated by the MAC-forgery (PRF-distinguishing) and key-guessing probabilities. Let the key space be $\mathcal{K}$, and the authentication tag length be b, then the probability of successfully forging data packets satisfies:

$$\Pr[Forge] \leq \frac{1}{|\mathcal{K}|} + 2^{-b} + \Pr[Nonce\ collision] \qquad (14)$$

Because the nonce and timestamp are both involved in key updates, attackers find it difficult to gain a sustained advantage by replaying old data packets. However, SPST-WSN do not claim to eliminate all security risks completely but rather improve attack costs, reduce leakage risks, and enhance anomaly auditability under resource-constrained conditions through lightweight mechanisms.

The framework still has residual risks. If a large number of neighboring sensor nodes are physically compromised at the same time, the attacker may influence both local aggregation and regional anomaly estimation. Therefore, SPST-WSN should be deployed together with device hardening, periodic key-material refresh, gateway redundancy, and maintenance inspection policies.

3.3. Discussion

The results demonstrate that security construction of urban water monitoring cannot rely solely on a single encryption scheme: AES-only offers lower latency but lacks privacy perturbation and anomaly routing; TLS-based systems enhance channel security but remain limited for large-scale access and edge auditing; Blockchain-based systems enhance evidence credibility but suffer latency, energy, and throughput losses; and DP-only systems reduce leakage risk but lose data utility. SPST-WSN's advantage lies in decomposing these mechanisms into lightweight edge modules that dynamically adjust security strength based on node size, risk, and sensitivity. The simulation data reflect urban water monitoring logic but have not yet incorporated long-term real-utility data; in practice, sensor drift, storm surges, communication obstruction, maintenance,

and seasonal variations could further affect performance. Future work could combine SPST-WSN with real river data, digital twin networks, and federated anomaly detection for cross-domain collaborative early warning without sharing raw data.

Although the simulation results support the scalability and security advantages of SPST-WSN, the current work remains simulation-based. Future deployment should validate the framework on real monitoring stations with heterogeneous sensors, intermittent links, seasonal water-quality fluctuations, and hardware-level energy constraints. Practical deployment should also specify gateway redundancy, maintenance intervals, secure key provisioning, and emergency fallback policies.

Federated multisensor fusion without centralizing raw data [23] is a concrete direction here. Practically, deployment needs out-of-band root-key provisioning with periodic (e.g. quarterly) refresh, N+1 gateway redundancy, attention to local data-retention rules, and a field pilot before production use, given the present results are simulation-based.

4. Conclusions

This study proposes SPST-WSN, a scalable distributed sensor network data privacy protection and secure transmission framework for urban water environment monitoring, built from four layers — sensor nodes, edge gateways, secure transmission, and cloud monitoring — integrating adaptive differential privacy, lightweight encryption, dynamic key updates, MAC-based integrity verification, edge security aggregation, and anomaly-aware routing. Experimental results show SPST-WSN maintains good latency, throughput, delivery rate, privacy protection, and detection performance even as node size, attack intensity, and privacy budget change, making it more suitable than Plain, AES-only, TLS-based, DP-only, and Blockchain-based methods for real-time large-scale urban water sensor networks. Future work could introduce real water data, trusted execution environments, lightweight blockchain auditing, and federated anomaly detection to improve engineering reliability and cross-domain collaboration.

References

[1] A. K. Agarwal, C. Mounaïm-Rousselle, P. Brequigny, A. Dhar, C. Hespel, C. Patel, D. K. Srivastava, G. Duraisamy, L. Le Moyne, N. Sharma, N. Labhasetwar, P. Singh, P. Das, P. K. Panigrahi, P. C. Shukla, P. Sakthivel, S. Mohan, S. Panigrahy, S. Sen, and H. Valera, (2025) *"Future of internal combustion engines using sustainable, scalable, and storable E-fuels and biofuels for decarbonizing transport and enabling advanced combustion technologies"* **Progress in Energy and Combustion Science *110***: 101236. DOI: https://doi.org/10.1016/j.pecs.2025.101236.

[2] A. O. Ali, O. Abdelrehim, M. M. Saafan, M. R. Elmarghany, and A. M. Hamed, (2025) *"Comprehensive review of battery management systems for electric vehicles: Thermal management, charging strategies, and emerging technologies"* **Journal of Power Sources *658***: 238269. DOI: https://doi.org/10.1016/j.jpowsour.2025.238269.

[3] P. Shrivastava, P. A. Naidu, S. Sharma, B. K. Panigrahi, and A. Garg, (2023) *"Review on technological advancement of lithium-ion battery states estimation methods for electric vehicle applications"* **Journal of Energy Storage *64***: 107159. DOI: https://doi.org/10.1016/j.est.2023.107159.

[4] C. Huang, Q. Shi, W. Ding, E. Q. Wu, and P. Mei, (2025) *"A Multirate-Fused State-of-Charge Estimation Scheme of Lithium-Ion Batteries for Electric Vehicles With Energy Harvesting Sensors"* **IEEE Transactions on Instrumentation and Measurement *74***: 1–11. DOI: 10.1109/TIM.2025.3547129.

[5] R. Guo and W. Shen, (2023) *"Lithium-Ion Battery State of Charge and State of Power Estimation Based on a Partial-Adaptive Fractional-Order Model in Electric Vehicles"* **IEEE Transactions on Industrial Electronics *70***(10): 10123–10133. DOI: 10.1109/TIE.2022.3220881.

[6] Q. Wang, C. Sun, and Y. Gu, (2023) *"Research on SOC estimation method of hybrid electric vehicles battery based on the grey wolf optimized particle filter"* **Computers and Electrical Engineering *110***: 108907. DOI: https://doi.org/10.1016/j.compeleceng.2023.108907.

[7] X. Lü, S. Li, X. He, C. Xie, S. He, Y. Xu, J. Fang, M. Zhang, and X. Yang, (2022) *"Hybrid electric vehicles: A review of energy management strategies based on model predictive control"* **Journal of Energy Storage *56***: 106112. DOI: https://doi.org/10.1016/j.est.2022.106112.

[8] J. J. Jui, M. A. Ahmad, M. I. Molla, and M. I. M. Rashid, (2024) *"Optimal energy management strategies for hybrid electric vehicles: A recent survey of machine learning approaches"* **Journal of Engineering Research *12***(3): 454–467. DOI: https://doi.org/10.1016/j.jer.2024.01.016.

[9] A. S. Mohammed, S. M. Atnaw, A. O. Salau, and J. N. Eneh, (2023) *"Review of optimal sizing and power management strategies for fuel cell/battery/super capacitor hybrid electric vehicles"* **Energy Reports *9***: 2213–2228. DOI: https://doi.org/10.1016/j.egyr.2023.01.042.

[10] I. Jarraya, F. Masmoudi, M. H. Chabchoub, and H. Trabelsi, (2019) *"An online state of charge estimation for Lithium-ion and supercapacitor in hybrid electric drive vehicle"* **Journal of Energy Storage *26***: 100946. DOI: https://doi.org/10.1016/j.est.2019.100946.

[11] X. Zhao, L. Wang, Y. Zhou, B. Pan, R. Wang, L. Wang, and X. Yan, (2022) *"Energy management strategies for fuel cell hybrid electric vehicles: Classification, comparison, and outlook"* **Energy Conversion and Management *270***: 116179. DOI: https://doi.org/10.1016/j.enconman.2022.116179.

[12] C. P. Sahwal, S. Sengupta, and T. Q. Dinh, (2024) *"Advanced Equivalent Consumption Minimization Strategy for Fuel Cell Hybrid Electric Vehicles"* **Journal of Cleaner Production *437***: 140366. DOI: https://doi.org/10.1016/j.jclepro.2023.140366.

[13] V. Duong, T. Thâm, W. Choi, and D.-W. Kim, (2016) *"State Estimation Technique for VRLA Batteries for Automotive Applications"* **Journal of Power Electronics *16***: 238–248. DOI: 10.6113/JPE.2016.16.1.238.

[14] M. Debbou and F. Colet. "Inductive wireless power transfer for electric vehicle dynamic charging". In: *2016 IEEE PELS Workshop on Emerging Technologies: Wireless Power Transfer (WoW)*. 2016, 118–122. DOI: 10.1109/WoW.2016.7772077.

[15] S. Kim, K. Kim, J. Ha, S. Kwon, and Y. Seo, (2016) *"A Study about Impact of Battery SOC on Fuel Economy of Conventional Diesel Vehicle"* **Transactions of the Korean Society of Automotive Engineers *24***: 480–486. DOI: 10.7467/KSAE.2016.24.4.480.

[16] A. A. Mahajan and N. P. Mungle, (2025) *"Fuzzy Logic Based Power Distribution Strategy for Hybrid Fuel Cell Vehicle"* **Metallurgical and Materials Engineering *31***(3): 358–367. DOI: 10.63278/1385.

[17] M. Averbukh, B. Rivin, and J. Vinogradov. "On-Board Battery Condition Diagnostics Based on Mathematical Modeling of an Engine Starting System". In: *SAE World Congress & Exhibition*. SAE International, 2007. DOI: https://doi.org/10.4271/2007-01-1476.

[18] C. Bharatiraja and G. Ramanathan. "A Hybridization of Photovoltaic and Battery Sources with Dual Input-Dual Output Converter for EV Charger". In: *Smart Grid Stability and Control*. Ed. by R. Krishan, D. R. Pullaguram, and S. R. Salkuti. Singapore: Springer Nature Singapore, 2025, 263–281. DOI: https://doi.org/10.1007/978-981-97-8634-3_18.

[19] C. Armenta-Déu, (2024) *"Battery Management for Improved Performance in Hybrid Electric Vehicles"* **Vehicles *6***(2): 949–966. DOI: 10.3390/vehicles6020045.

[20] R. R. Kumar, C. Bharatiraja, K. Udhayakumar, S. Devakirubakaran, K. S. Sekar, and L. Mihet-Popa, (2023) *"Advances in Batteries, Battery Modeling, Battery Management System, Battery Thermal Management, SOC, SOH, and Charge/Discharge Characteristics in EV Applications"* **IEEE Access *11***: 105761–105809. DOI: 10.1109/ACCESS.2023.3318121.

[21] G. Vuylsteke, H. Wu, W. Moore, and D. Washington. "Impact of Battery Aging on the State of Charge-Open Circuit Voltage Relationship and Its Effects on Battery Capacity Estimation". In: *WCX SAE World Congress Experience*. SAE International, 2025. DOI: https://doi.org/10.4271/2025-01-8558.

[22] K. W. E. Cheng, B. P. Divakar, H. Wu, K. Ding, and H. F. Ho, (2011) *"Battery-Management System (BMS) and SOC Development for Electrical Vehicles"* **IEEE Transactions on Vehicular Technology *60***: 76–88. DOI: https://doi.org/10.1109/TVT.2010.2089647.

[23] M. Wang and T. Huang. "An Integrated Electric Energy Management System to Improve Fuel Economy". In: *Proceedings of the FISITA 2012 World Automotive Congress*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, 115–122. DOI: https://doi.org/10.1007/978-3-642-33829-8_12.

[24] J. Wishart, M. Shirk, T. Gray, and N. Fengler. "Quantifying the Effects of Idle-Stop Systems on Fuel Economy in Light-Duty Passenger Vehicles". In: *SAE 2012 World Congress & Exhibition*. SAE International, 2012. DOI: https://doi.org/10.4271/2012-01-0719.

[25] I. Cho and J. Lee, (2020) *"Characteristics of Battery SOC According to Drive Output and Battery Capacity of Parallel Hybrid Electric Vehicle"* **Applied Sciences *10***(8): DOI: 10.3390/app10082833.

[26] P. Ruetschi, (2004) *"Aging mechanisms and service life of lead–acid batteries"* **Journal of Power Sources *127***(1): 33–44. DOI: https://doi.org/10.1016/j.jpowsour.2003.09.052.

[27] S. Singirikonda and Y. P. Obulesu. "Advanced SOC and SOH Estimation Methods for EV Batteries—A Review". In: *Advances in Automation, Signal Processing, Instrumentation, and Control*. Ed. by V. L. N. Komanapalli, N. Sivakumaran, and S. Hampannavar. Singapore: Springer Nature Singapore, 2021, 1963–1977.

[28] Y. Wang, J. Tian, Z. Sun, L. Wang, R. Xu, M. Li, and Z. Chen, (2020) *"A comprehensive review of battery modeling and state estimation approaches for advanced battery management systems"* **Renewable and Sustainable Energy Reviews *131***: 110015. DOI: https://doi.org/10.1016/j.rser.2020.110015.

[29] J. S. Kim, J. W. Kim, J. H. Jeong, S. C. Jeong, and J. W. Lee, (2016) *"Effects of Initial SOC of 270-Volt Battery on Operating Performance of Gasoline Engine and Electric motor in a Parallel Hybrid Vehicle under IM240 Driving Cycle Mode"* **Advances in Automobile Engineering *2016***: 1–9. DOI: 10.4172/2167-7670.S1-008.

[30] United Nations. *Global technical regulation on worldwide harmonized light vehicles test procedure*. Tech. rep. ECE/TRANS/180/Add.15. Global Technical Regulation No. 15 (GTR 15) on Worldwide harmonized Light vehicles Test Procedures (WLTP), established in the Global Registry on 15 November 2017[reference:0]. United Nations, 2017.